

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-008906

(43)Date of publication of application : 10.01.2003

.....
(51)Int.Cl. H04N 1/41
H03M 7/30
H04N 7/24

.....
(21)Application number : 2001-192670 (71)Applicant : MITSUBISHI ELECTRIC
CORP

(22)Date of filing : 26.06.2001 (72)Inventor : TANIGUCHI MASAHIRO

.....
(54) PROCESSING METHOD FOR BIT MODELING AND PROCESSING CIRCUIT
USING IT

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a processing method for bit modeling at a fast processing speed and a processing circuit using it.

SOLUTION: A context and a decision changing with a state of a significant flag and a code bit of a bit being a processing object and a surrounding bit group are simultaneously generated according to the sig (significance propagation decoding) path, the context and the decision of the code bit are adopted only when a value of the processing object bit is '1', the significant flag is updated, the decision of the code bit are aborted when the value of the processing object bit is '0', the processing circuit for the bit modeling to update the processed flag is applied to bits circle 0 - circle 3 of the same

group at the same time to process them in parallel independently of whether the value of the processing object bit is '1' or '0'.

LEGAL STATUS [Date of request for examination]
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

* NOTICES *

JP0 and INPIT are not responsible for any damages caused by the use of this translation.

1.This document has been translated by computer. So the translation may not reflect the original precisely.

2.**** shows the word which can not be translated.

3.In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] In the art of the SHIGUNIFIKANSU propagation decoding pass (Significance propagation decoding pass) of the bit modeling of JPEG2000 The context and decision of the data which change according to the condition of the significant flag of the bit which is a processing object, and a surrounding bit group, The context and decision of a sign bit which change according to the condition of the bit which is said processing object, and the sign bit of a surrounding bit group are generated to coincidence. Only when the value of the bit which is said processing object is 1, the context and decision of said sign bit are adopted. Update said significant flag, and when the value of the bit which is said processing object is 0, it discards. The art of the bit modeling to which the value of the bit which is said processing object is characterized by applying at coincidence the processing circuit of the bit modeling which updates a processed flag irrespective of 1 or 0 to four bits of the same group, and

processing it to juxtaposition.

[Claim 2] In the art of magnitude refinement MENTOPASU (Magnitude refinement pass) of the bit modeling of JPEG2000 The 2nd bit, a processed flag, and a significant flag are referred to. it is the information on whether the bit which is a processing object is processed for the first time by said magnitude refinement MENTOPASU -- significant -- When judging whether it processes or not and processing by said magnitude refinement MENTOPASU The art of the bit modeling characterized by generating the context and decision of a bit which are said processing object, applying at coincidence the processing circuit of the bit modeling which also updates said processed flag to four bits of the same group, and processing to juxtaposition.

[Claim 3] In the art of the clean-up pass (Cleanup pass) of the bit modeling of JPEG2000 It judges whether all the bits of the same group who is a processing object can process collectively, when unsettled. The 1st processing circuit of said bit modeling where all said same group's bit groups generate a context and decision special at the time of un-significant **, Nothing is performed about a bit [finishing / processing], but the 2nd processing circuit of said bit modeling which processes the bit of non-**** is prepared. One of said the 1st processing circuit

The art of the bit modeling characterized by applying said 2nd processing circuit to four bits of said same group at coincidence, and processing to juxtaposition.

[Claim 4] The art of the bit modeling characterized by processing three kinds of coding pass in order about the same bit plane in the art of the bit modeling of JPEG2000.

[Claim 5] The art of the bit modeling characterized by processing to juxtaposition three groups whom the same bit plane adjoined for three kinds of every coding pass in the art of the bit modeling of JPEG2000.

[Claim 6] The art of the bit modeling characterized by processing to two or more juxtaposition with the same bit plane in the art of bit modeling according to claim 5.

[Claim 7] The art of the bit modeling characterized by processing to juxtaposition with two or more bit planes in the art of one of bit modeling according to claim 5 or 6.

[Claim 8] The processing circuit characterized by having the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag and sign bit of the bit which is a processing object, and a surrounding bit group, and the register which stores the processed

flag of the bit which is said processing object in the processing circuit using the art of bit modeling according to claim 1.

[Claim 9] in the processing circuit using the art of bit modeling according to claim 2, it is the information on whether the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag of the bit which is a processing object, and a surrounding bit group, and the bit which is a processing object are processed for the first time by magnitude refinement MENTOPASU -- significant -- the processing circuit characterized by having the register which stores the 2nd bit.

[Claim 10] the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- the data bit for a code block size, a sign bit, a processed flag, and a significant flag -- and significant -- the processing circuit characterized by having the register which stores the 2nd bit.

[Claim 11] the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- a data bit, a sign bit, a processed flag, and a significant flag -- and significant -- the processing circuit characterized by having the register which stores only a part for the bit which sets the 2nd bit as the object of processing.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the processing circuit which uses

the art of bit modeling, and it, especially is a processing circuit using the art of bit modeling and it about improvement in the speed of the bit modeling in coding of JPEG2000.

[0002]

[Description of the Prior Art] JPEG2000 (it defines by ISO/IEC FDIS 15444-1.)

Hereafter, it is indicated as JPEG2000. Coding processing expresses a quantization multiplier with a sign bit and an absolute value. A context and decision are generated, investigating the relation of the four directions of each bit for the bit which expresses an absolute value with the unit of a code block in order of a high order bit plane to a low order bit plane (it is hereafter indicated as bit modeling.).

[0003] Moreover, when the bit plane all whose bits are 0 continues zero N from the most significant showing the absolute value within a code block of a bit plane, it processes independently.

[0004] For example, there is precision of M bit plane as an absolute value, and in a certain code block, when 0 bit of high orders N of all absolute values is 0, only $N=M-N_0$ bit plane is set as the object of bit modeling.

[0005] The top bit plane set as the object of bit modeling is called MSB (it is

indicated as MSB a MostSignificant Bit plane and the following.), and the lowest bit plane is called LSB (it is indicated as LSB a Less Significant Bit plane and the following.).

[0006] From MSB to LSB, for every bit plane, the data within a code block are processed and go. In each bit plane, grouping of the four bits is collectively carried out to the lengthwise direction, and bit modeling is performed in order of a raster.

[0007] Drawing 14 shows the sequence (0-23) of processing the group when performing bit modeling with the code block size of the width 8(0-7) x length 12 (0-11).

[0008] moreover, drawing 15 shows the order (0-7) of processing of each bit (a lengthwise direction -- every four bits 4 (0-3), for example, width 0x length, and the width 1x length 4 (0-3)) over the group of a bit plane.

[0009] In bit modeling, it is Significance. propagationdecoding pass (it is hereafter indicated as sig pass.), Magnitude refinement pass (it is hereafter indicated as ref pass.), Cleanup There is three sorts of coding pass called pass (it is hereafter indicated as cln pass.), and it is processed in order of sig pass, ref pass, and cln pass.

[0010] That is, this processing evaluates a bit in a certain bit plane in drawing 14 and the order shown in 15, if it is data which it should process with sig pass, it will process it with sig pass, otherwise, it progresses to the following bit.

[0011] After code block processing is completed, next, it evaluates in order as ref pass, and goes. Finally cln pass is processed. If a certain bit is processed with the early (a priority is high) coding pass of processing sequence at this time, that bit will not be processed with other coding pass.

[0012] Therefore, each bit will be processed with one kind of coding pass in the three above-mentioned kinds. Although this processing is repeated from MSB to LSB, it restricts to MSB and processes only with cln pass.

[0013] For example, the operation of the bit modeling of x (7 bit-plane $\times 3$ coding pass +1 coding pass) (8x12) code block size is needed at the time of $N=8$ bits.

[0014]

[Problem(s) to be Solved by the Invention] In the art of the above bit modeling of conventional JPEG2000, when carrying out the bit modeling of a certain code block, according to the bit plane which processes one data, processing of $x(N-1)3+1$ time is needed, for example.

[0015] This has the problem that processing speed becomes slow, compared

with the quantization whose one data requires only one processing irrespective of bit precision.

[0016]

[Means for Solving the Problem] The processing circuit using the art of bit modeling and it concerning this invention In the art of the SHIGUNIFIKANSU propagation decoding pass (Significance propagation decoding pass) of the bit modeling of JPEG2000 The context and decision of the data which change according to the condition of the significant flag of the bit which is a processing object, and a surrounding bit group, The context and decision of a sign bit which change according to the condition of the bit which is a processing object, and the sign bit of a surrounding bit group are generated to coincidence. Only when the value of the bit which is a processing object is 1, the context and decision of a sign bit are adopted. A significant flag is updated, when the value of the bit which is a processing object is 0, it discards, and irrespective of 1 or 0, the value of the bit which is a processing object applies at coincidence the processing circuit of the bit modeling which updates a processed flag to four bits of the same group, and processes it to juxtaposition.

[0017] Moreover, it sets to the art of magnitude refinement MENTOPASU

(Magnitude refinement pass) of the bit modeling of JPEG2000. The 2nd bit, a processed flag, and a significant flag are referred to. It is the information on whether the bit which is a processing object is processed for the first time by magnitude refinement MENTOPASU -- significant -- When judging whether it processes or not and processing by magnitude refinement MENTOPASU The context and decision of a bit which are a processing object are generated, and the processing circuit of the bit modeling which also updates a processed flag is applied to four bits of the same group at coincidence, and is processed to juxtaposition.

[0018] Moreover, it sets to the art of the clean-up pass (Cleanup pass) of the bit modeling of JPEG2000. It judges whether all the bits of the same group who is a processing object can process collectively, when unsettled. The 1st processing circuit of bit modeling where all the same group's bit groups generate a context and decision special at the time of un-significant **, Nothing is performed about a bit [finishing / processing], but the 2nd processing circuit of the bit modeling which processes the bit of non-**** is prepared, the 2nd processing circuit is applied to one of the 1st processing circuit, and four bits of the same group at coincidence, and it processes to juxtaposition.

[0019] Moreover, in the art of the bit modeling of JPEG2000, three kinds of coding pass is processed in order about the same bit plane.

[0020] Moreover, in the art of the bit modeling of JPEG2000, three groups whom the same bit plane adjoined are processed to juxtaposition for three kinds of every coding pass.

[0021] Moreover, in the art of bit modeling according to claim 5, it processes to two or more juxtaposition with the same bit plane.

[0022] Moreover, in the art of one of bit modeling according to claim 5 or 6, it processes to juxtaposition with two or more bit planes.

[0023] Moreover, in the processing circuit using the art of bit modeling according to claim 1, it has the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag and sign bit of the bit which is a processing object, and a surrounding bit group, and the register which stores the processed flag of the bit which is a processing object.

[0024] moreover, in the processing circuit using the art of bit modeling according to claim 2, it is the information on whether the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag of the bit which is a processing object, and a surrounding bit

group, and the bit which is a processing object are processed for the first time by magnitude refinement MENTOPASU -- significant -- it has the register which stores the 2nd bit.

[0025] moreover, the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- the data bit for a code block size, a sign bit, a processed flag, and a significant flag -- and significant -- it has the register which stores the 2nd bit.

[0026] furthermore, the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- a data bit, a sign bit, a processed flag, and a significant flag -- and significant -- it has the register which stores only a part for the bit which sets the 2nd bit as the object of processing.

[0027]

[Embodiment of the Invention] This invention is explained below gestalt 1. of operation. Drawing 1 is drawing having shown the case where bit group O0-O3 of the group in a bit plane with JPEG2000 by the gestalt 1 of operation were processed. When processing the group who shows by bit group O0-O3 with sig pass with reference to drawing 1 , the information about the bit shown by x among the bits of the perimeter is needed.

[0028] Drawing 2 is drawing having shown the bit which is needed when processing the bit of O0 in drawing 1 . With reference to drawing 2 , the bit processed ahead of bit O0 in processing of the same coding pass is shown about x0, x1, x2, x3, and x5. The bit processed after bit O0 is shown about **4, **6, and **7.

[0029] The conditions from which a bit serves as a processing object in sig pass are that the bit is insignificance (it is indicated hereafter that it is un-significant.), and a surrounding bit is one or more piece significance (it is indicated hereafter that it is significant.).

[0030] Therefore, the information (it is hereafter indicated as a significant flag.) on whether the bit which is a processing object, and a surrounding bit are significant is needed. The significant flag is stored in the register.

[0031] If bit O0 is un-significant and one or more of bit group x0-**7 of a perimeter pieces are significant, it will process with sig pass.

[0032] According to the condition of the surrounding significant flag of bit group x0-**7, a context (it is hereafter indicated as the context of data.) is generated, and let the value of bit O0 itself be decision (for it to be hereafter indicated as the decision of data.).

[0033] If the value of bit O0 is 1, since the data is un-significant and it will become significant, a significant flag is updated and bit modeling is further performed also about a sign bit.

[0034] The sign bit of bit O0 and the surrounding sign bit of bit group x0-^{**}7 are needed for sign bit processing.

[0035] According to the condition of the surrounding sign bit of bit group x0-^{**}7, a context (it is hereafter indicated as the context of a sign bit.) is generated, and let the result of the XOR operation of the value according to the sign bit and context of bit O0 be decision (for it to be hereafter indicated as the decision of a sign bit.).

[0036] The context / decision of data, and the context/decision of a sign bit are generated to coincidence, and only when required (the value of bit O0 is 1), it adopts the context/decision of a sign bit.

[0037] Moreover, if it discards when there is no need (the value of bit O0 is 0), it can process to juxtaposition.

[0038] Moreover, when processed with sig pass irrespective of whether it is [0] whether the value of bit O0 is 1, the information (it is hereafter indicated as a processed flag.) that it is processing settled in the bit plane is updated.

[0039] When processing a certain bit with sig pass, the significant flag and sign

bit of the value of the bit, and its bit and surrounding bit are referred to.

[0040] That is, the register which stores the significant flag and sign bit of bit group $x0-^{**}7$ of the value of bit O0, and bit O0 and a perimeter is needed.

[0041] Moreover, it is necessary to prepare also about the register for the processed flag of bit O0.

[0042] Drawing 3 is drawing having shown the bit which is needed when processing the bit of O1 in drawing 1 . With reference to drawing 3 , the surrounding significant flag of bit group $x0-^{**}7$ is referred to for decision whether bit O1 is processed with sig pass.

[0043] Moreover, since the significant flag of bit x1 is the significant flag and EQC of bit O0 of drawing 2 , processing of bit O1 cannot be started until it may be updated at the time of processing of bit O0 and processing of bit O0 is completed, but if it turns out whether the significant flag of bit x1 is updated when processing bit O1, it will become possible to process to coincidence.

[0044] That is, at the time of un-significant **, the value is 1, and when one or more bit groups of the perimeter are significant, bit x1 judges that it becomes significant by processing of the last sig pass, and it processes bit O0 and bit O1 to coincidence by considering as the significant flag after updating at

coincidence with reference to the significant flag of bit x1 and the bit group of the perimeter, and the value of the bit.

[0045] Drawing 4 is drawing having shown the bit which is needed when processing the bit of O2 in drawing 1 . With reference to drawing 4 , the surrounding significant flag of bit group x0-^{**}7 is referred to for decision whether bit O2 are processed with sig pass.

[0046] Since the significant flag of bit x1 is the significant flag and EQC of bit O1 of drawing 3 , processing of bit O2 cannot be started until it may be updated at the time of processing of bit O1 and processing of bit O1 is completed, but if it turns out whether the significant flag of bit x1 is updated when processing bit O2, it will become possible to process to coincidence.

[0047] That is, at the time of un-significant ^{**}, the value is 1, and when one or more bit groups of the perimeter are significant, bit x1 judges that it becomes significant by processing of the last sig pass, and it processes bit O1 and bit O2 to coincidence by considering as the significant flag after updating at coincidence with reference to the significant flag of bit x1 and the bit group of the perimeter, and the value of the bit.

[0048] Drawing 5 is drawing having shown the bit which is needed when

processing the bit of O3 in drawing 1 . With reference to drawing 5 , the surrounding significant flag of bit group x0-^{**}7 is referred to for decision whether bit O3 are processed with sig pass.

[0049] Since the significant flag of bit x1 is the significant flag and EQC of bit O2 of drawing 4 , processing of bit O3 cannot be started until it may be updated at the time of processing of bit O2 and processing of bit O2 is completed, but if it turns out whether the significant flag of bit x1 is updated when processing bit O3, it will become possible to process to coincidence.

[0050] That is, at the time of un-significant ^{**}, the value is 1, and when one or more bit groups of the perimeter are significant, bit x1 judges that it becomes significant by processing of the last sig pass, and it processes bit O2 and bit O3 to coincidence by considering as the significant flag after updating at coincidence with reference to the significant flag of bit x1 and the bit group of the perimeter, and the value of the bit.

[0051] According to the gestalt 1 of this operation, the parallel processing of the group of bit group O0-O3 becomes possible, and it judges whether 1-bit data are processed with sig pass, and it becomes possible to constitute the circuit which outputs 1/2 0 / set of a context/decision.

[0052] Moreover, it becomes possible to constitute the circuit which outputs the set of 0-8 a context/decision about one group (four data) from arranging in four-piece juxtaposition the circuit which processes 1-bit data, and processing bit O0, O1, O2, and O3, respectively.

[0053] Gestalt 2. of operation, next the parallel processing approach by the gestalt 2 of operation are described. Since it is the gestalt of the operation which processes drawing 1 -5 with ref pass, this is explained using the same drawing as drawing 1 -5 in the gestalt 1 of operation.

[0054] First, when processing the group who shows by bit group O0-O3 of drawing 1 with ref pass, the information about the bit shown by x among the groups of the perimeter is needed. when this group is further divided for every bit, it is shown in drawing 2 -5 -- each -- the information on 8 bits of perimeters is needed about bit O0, O1, O2, and O3.

[0055] The bit is not processed with sig pass and the processing conditions of the bit in ref pass need a significant thing. Therefore, the bit which is a processing object is processing settled, or the information on whether to be significant or not is needed.

[0056] In bit O0 of drawing 2 , if it is unsettled and significant, it will process with

ref pass. When bit O0 is processed with ref pass for the first time, the context of data is generated according to the condition of the surrounding significant flag of bit group x0-^{**7}, when that is not right, it is not based on the condition of a significant flag, but the context of the data of immobilization is generated.

[0057] Updating a processed flag, when the value of bit O0 itself is made into the decision of data also at the time of any and it is processed with ref pass, the flag processed [a significant flag and] uses the same thing as sig pass. furthermore, the information (hereafter significant -- it is indicated as the 2nd bit.) on whether bit O0 is processed with ref pass for the first time is stored in the register.

[0058] Refer to the processed flag and significant flag of bit O1 for decision whether in drawing 3 , bit O1 is processed with ref pass. Similarly, the processed flag and significant flag of bit O2 and bit O3 are referred to also about drawing 4 and 5, respectively.

[0059] moreover, each bit O shown in drawing 2 -5 -- the value of bit group O0-O3, the processed flag and significant flag which are shown for processing 0, O1, O2, and O3 at drawing 1 , and a significant flag with bit O and bit x are needed.

[0060] Since ref pass does not update a significant flag according to the gestalt 2

of this operation, the parallel processing of the group of bit group O0-O3 becomes possible, and it judges whether 1-bit data are processed with ref pass, and it becomes possible to constitute the circuit which outputs 0/1 set of a context/decision.

[0061] Moreover, it becomes possible to constitute the circuit which outputs the set of 0-4 a context/decision about one group (four data) from arranging in four-piece juxtaposition the circuit which processes 1-bit data, and processing bit O0, O1, O2, and O3, respectively.

[0062] The parallel processing approach by the gestalt 3 of gestalt 3. implementation of operation is described. Since it is the gestalt of the operation which processes drawing 1 -5 with cln pass, this is explained using the same drawing as drawing 1 -5 in the gestalt 1 of operation.

[0063] First, when processing the group who shows by bit group O0-O3 of drawing 1 with cln pass, the information about the bit shown by x among the groups of the perimeter is needed. when this group is further divided for every bit, it is shown in drawing 2 -5 -- each -- the information on 8 bits of perimeters is needed about bit O0, O1, O2, and O3.

[0064] The processing conditions of the bit in cln pass are that the bit is unsettled.

However, in MSB, since only cln pass is processed, all bits serve as a processing object. Therefore, the information on whether the bit which is a processing object is processing settled is needed.

[0065] moreover, cln pass shows to drawing 2 -5 -- each -- in bit O0, O1, O2, and O3, altogether, when unsettled, it judges whether it can process collectively. each bit O shown in drawing 2 -5 -- 0, O1, O2, and O3 are alike, respectively, it sets, and all bit group x0-**7 of a perimeter generate a special context (it is hereafter indicated as the context of a run length.) at the time of un-significant **.

[0066] moreover, it is shown in coincidence drawing 2 -5 -- each -- if all of the value of bit O0, O1, O2, and O3 are 0, decision (it is hereafter indicated as the decision of a run length.) 0 will be generated, and processing of the group will be ended.

[0067] moreover, it is shown in drawing 2 -5 -- each -- the value of bit O0, O1, O2, and O3 will set decision of a run length to 1, if at least one is 1. at this time, it is shown in drawing 2 -5 -- each -- 2-bit data express the location of the bit which has the value of 1 in the beginning among bit O0, O1, O2, and O3.

[0068] That is, if bit O0 is 1 of the beginning and it is 00 and bit O1, it will be similarly referred to as 10 and 11 O1 and the following. These two data are

generated with the context (it is hereafter indicated as the context of UNIFORM.) following a run length, and are outputted as decision (it is hereafter indicated as the decision of UNIFORM.), respectively.

[0069] When [all] only the context of a run length is outputted, since it is as [un-significant **], renewal of a significant flag is not performed.

[0070] Since the bit of the location which the decision of UNIFORM shows is un-significant and it becomes significant when the context of UNIFORM is also outputted, the significant flag of the bit is updated.

[0071] However, since it expresses that the bit of the location which the decision shows in the context of UNIFORM itself is 1, the context of the sign bit of the bit is generated like sig pass.

[0072] The generation method of the context of the sign bit in cIn pass is the same as that of sig pass, when the decision of UNIFORM is 11, it is ended by processing of only a sign bit, but when other, it processes 1 bit of bits which remain at a time.

[0073] The approach of processing of the bit which remains is the same as that of sig pass except for processing, even when all the bits around the bit are un-significant. Moreover, since the bit before the location which the decision of

UNIFORM shows is as [un-significant **], renewal of a significant flag is not performed.

[0074] According to the gestalt 3 of this operation, when not becoming the processing object of the context of a run length, an unsettled thing is processed in order among the bits in that group. Moreover, about the unsettled bit when not becoming the processing object of the bit after the context of UNIFORM, or the context of a run length, in order to perform the same processing as sig pass, it becomes possible to constitute the same circuit as sig pass.

[0075] Moreover, although sig pass constitutes the circuit which outputs the set of 0-8 a context/decision from processing to juxtaposition that which each bit became independent of According to the cln pass, the context/decision of further 0 run length, Juxtaposition is made to process the circuit which processes the context/decision of UNIFORM. The number of bits processed with ->cln pass when the context of the number of bits and 0 run length after the context of ->UNIFORM is not included, including the context of UNIFORM is taken into consideration. When the context of zero run length is included -> - Zero run length, UNIFORM, It becomes possible to constitute the circuit which outputs the set of 0-10 a context/decision from calculating the context/decision of each bit.

[0076] Although the circuitry according to individual is described [gestalten / 1-3 / of gestalt 4. implementation of operation] about each coding pass, with the gestalt 4 of operation, the same group in the same bit plane is processed to coincidence in the circuit of the gestalten 1-3 of operation.

[0077] since [in this case,] three kinds of coding pass needs to share information -- the significant flag for a code block size -- significant -- it is necessary to store the 2nd bit and a processed flag in a register

[0078] Moreover, about a sign bit and the data bit of the bit plane used as a processing object, to read a part for a code block size, and the part which actually serves as a processing object for every coding pass, it is necessary to store only the part in a register.

[0079] According to the gestalt 4 of this operation, compared with the gestalten 1-3 of operation, processing speed can be made quick.

[0080] Gestalt 5. drawing 6 of operation is drawing showing the range of the data used when processing three groups by the gestalt 5 of operation who continued. With the gestalt 4 of operation, it was the approach of processing the same group in the same bit plane to coincidence with three kinds of coding pass.

[0081] However, although it processes in order of sig pass ->ref pass ->cln pass

and goes about a certain bit plane within a code block, each bit needs to process each coding pass in order, if it is going to obtain exact bit modeling by constraint of updating a significant flag with two or more pass, such as being processed only with one coding pass, and sig pass or cln pass.

[0082] That is, in a certain bit plane, although four bits are processed to coincidence, it will process in order with three kinds of coding pass in fact, and it is necessary to press down the clock frequency as a circuit low. In order to solve this, with the gestalt 5 of operation, the approach of processing the group (bit group O0-O3, bit group O4-O7, bit group O8-O11) who followed the longitudinal direction with respectively different coding pass is taken.

[0083] Drawing 7 is drawing showing the range of the data used when processing the group of bit group O8-O11. Moreover, drawing 8 is drawing showing the range of the data used when processing the group of bit group O4-O7. Moreover, drawing 9 is drawing showing the range of the data used when processing the group of bit group O0-O3.

[0084] sig pass is coding pass processed when a significant bit exists in one or more surrounding bits un-intentionally, is performed first and can be processed from other two pass, without receiving no constraint.

[0085] It is the coding pass which processes bit group O8-O11 into three groups who show by drawing 6 , and drawing 7 is the use range of the data. Since bit group O8-O11 are processed with sig pass, the processed flag of bit group O8-O11 and a significant flag are updated by conditions.

[0086] Bit group x5-x8 in drawing 6 -9, bit group O0-O3, and bit group O4-O7 are already processing settled by sig pass.

[0087] Next, ref pass is coding pass which already processes significant data, and it is necessary to take into consideration whether it is significant at the time of processing.

[0088] Although a significant flag is referred to, it cannot be processed whether the bit is already significant, when the bit is processed with the sig pass of the same bit plane and becomes significant.

[0089] That is, since it is necessary to judge whether it was processed with sig pass, it cannot process to sig pass and coincidence.

[0090] It is the coding pass which processes bit group O4-O7 into three groups who show by drawing 6 , and drawing 8 is the use range of the data. Since bit group O4-O7 are processed with ref pass, with reference to the processed flag of bit group O4-O7, and a significant flag, a processed flag is updated according

to conditions.

[0091] ref pass can refer the flag of bit group O4-O7 after updating with sig pass by processing bit group O8-O11 with sig pass, and processing bit group O4-O7 with ref pass.

[0092] Moreover, although the group who processes with sig pass and ref pass showed the configuration 1 group ** made into the longitudinal direction with the gestalt 5 of this operation in order to make a circuit scale into min, two or more groups may shift.

[0093] Drawing 6 , bit group 5x-x8 in 8 and 9, and bit group O0-O3 are already processing settled by ref pass.

[0094] Moreover, cIn pass is un-significant and is coding pass which processes the data which were not processed with sig pass. With the gestalt 5 of this operation, since it is all unsettled data, the data which significant data are already processed with ref pass, and were not processed with the inner sig pass of un-significant data process only unsettled data with reference to a processed flag.

[0095] Since it is necessary to refer to the significant flag of the bit used as a processing object, and a perimeter bit, it cannot process to sig pass and

coincidence.

[0096] It is the coding pass which processes bit group O0-O3 into three groups who show by drawing 6 , and drawing 9 is the use range of the data. Since bit group O0-O3 are processed with cln pass, with reference to the processed flag of the significant flag of bit group O4-O7, and bit group O0-O3, a significant flag is updated according to conditions.

[0097] Moreover, bit group x5-x8 in drawing 6 and 9 are already processing settled by cln pass.

[0098] cln pass can refer bit group O0-O3 after updating with sig pass and ref pass, and the flag of bit group O4-O7 by processing bit group O4-O7 with bit group O8-O11 and ref pass, and processing bit group O0-O3 with cln pass with sig pass.

[0099] Moreover, although the group who processes with sig pass and ref pass showed the configuration 1 group ** made into the longitudinal direction with the gestalt 5 of this operation in order to make a circuit scale into min, the group number processed with sig pass, ref pass, and cln pass is arbitrary, and good.

[0100] When processing three kinds of coding pass to juxtaposition with the above-mentioned configuration, in accordance with the same approach as the

gestalten 1-4 of operation, only three kinds of coding pass can process four data to juxtaposition.

[0101] Moreover, it can process to continuation by shifting the group who processes by bit group **9-**12 with sig pass, and processes like bit group O4-O7 with bit group O8-O11 and cln pass with ref pass in drawing 6 .

[0102] Although each coding pass outputs the set of 0-10 a context/decision, it is necessary to store these data for every coding pass.

[0103] according to the gestalt 5 of this operation -- the significant flag for a code block size -- significant -- what is necessary is to store in the register only the part set as the object of processing, when actually processing a sign bit, the data bit of the bit plane used as a processing object, and a processed flag in order to process three kinds of coding pass at a time although it is necessary to store bit [2nd] information in a register

[0104] Moreover, when spacing of the group who processes with each coding pass is made into one group like drawing 6 , the part set as the object of processing becomes 20 bits of x5-x8, O0-O11, and **9-**12 with 30 bits, the data bit, and the processed flag of O**x by the sign bit.

[0105] Therefore, compared with the gestalten 1-4 of operation, it can process at

a high speed and a circuit scale can be reduced.

[0106] Gestalt 6. of operation, next the parallel processing approach by the gestalt 6 of operation are described. a significant flag [in / in this / drawing 6] -- and significant -- it is the approach of performing the 2nd bit in the amount of registers of only the part (20 bits (significant the 2nd bit) of 30 bits (significant flag) of O^{**x} , $x5-x8$, $O0-O11$, and $**9-^{**12}$) made into the object of processing.

[0107] It is because it needs to share information that a significant flag is the code block-size part need while carrying out bit modeling from MSB to LSB, so it is necessary to hold.

[0108] If the condition after processing to the last bit plane is [that the condition after processing with the bit plane in front of that should just be known] calculable beforehand, it will become unnecessary however, to share information between bit planes in processing of a certain bit plane.

[0109] With the gestalt 6 of operation, the value of the data which it is going to process is investigated and the approach of judging by whether the bit whose value is 1 is in a high order from the bit plane used as a processing object is taken.

[0110] Drawing 10 is drawing in which having divided into the sign bit and the

absolute value one certain data which it is going to process by bit modeling, and having shown the value. With reference to drawing 10 , from MSB, since all the values to the bit on one piece are 0, the significant flag before sig pass processing of this data presupposes that it is un-significant from the bit plane which is a processing object.

[0111] Moreover, in drawing 11 , from MSB, from the bit plane which is a processing object, since the value to the bit on one piece is 1, the significant flag before sig pass processing of this data presupposes that it is significant.

[0112] If this is performed when reading a data bit for every bit plane, since it will become unnecessary to share information between bit planes, the number of registers of a significant flag is reducible.

[0113] Therefore, a significant flag should just have a register by 30 bits of $O^{**}x$ shown in drawing 6 .

[0114] It becomes unnecessary moreover, to share information between bit planes by calculating intentionally similarly [the 2nd bit] beforehand the bit plane processed with ref pass for the first time, i.e., the next bit plane which became significant.

[0115] With the gestalt 6 of this operation, the data which it is going to process

are investigated and the value of a 1-bit high order judges by whether all the bits of a high order are 0 from it from the bit plane used as a processing object 1.

[0116] more significant than the bit plane of a processing object, since all the values of the bit of a high order are 0, when drawing 10 is taken for an example -- it does not become the 2nd bit. moreover, in drawing 11 , since all the values of a high order bit are 0 from it in 1, the value of the 1-bit high order of the bit plane which carries out a processing object is significant -- it becomes the 2nd bit.

[0117] significant, since it will become unnecessary to share information between bit planes, if this is performed when reading a data bit for every bit plane -- the number of registers which is the 2nd bit is reducible.

[0118] therefore, significant -- what is necessary will be just to have the 2nd bit by 20 bits of x5-x8 shown in drawing 6 , 00-011, and **9-**12

[0119] The gestalt 6 of operation is a circuit which processes three kinds of coding pass to juxtaposition in the same bit plane, and let it be the basic circuit of the gestalt of the following operations.

[0120] gestalt 7. of operation -- the gestalt 7 of this operation is an art which uses two or more circuits of the gestalt 6 of operation in the same bit plane. Drawing

13 is drawing showing the range of the data used when processing three groups who continued to two or more juxtaposition. This uses bit group O000-O011 and the circuit stated with the gestalt 6 of two operations about O100-O111.

[0121] With reference to drawing 13 , bit group O000-O011 are processed as they are using the same circuit as the gestalt 6 of operation. Moreover, since bit group x100-x104 are data which should be processed before original bit group O100-O111, they need to calculate a significant flag beforehand.

[0122] Although the gestalt 6 of operation showed the count approach of the significant flag to the bit plane of the 1-bit high order of the bit plane used as a processing object, with the gestalt 7 of this operation, the information after processing of the group on one step is further needed for a lengthwise direction. In addition to the approach shown with the gestalt 6 of operation, this is calculable by taking into consideration the data of the bit plane of a processing object.

[0123] The case where x100 of drawing 13 have a value like drawing 10 -12 in below is described.

[0124] In drawing 10 , to the bit plane which serves as a processing object from MSB, since all values are 0, it is un-significant. In drawing 11 , since there is a bit

whose value is 1 by the 1-bit high order of the bit plane which serves as a processing object from MSB, it is significant.

[0125] Moreover, in drawing 12 , since there is a bit (in this case, bit plane of a processing object) whose value is 1 even in the bit plane which serves as a processing object from MSB, it is significant. Thus, even the group on one step becomes possible [calculating a significant flag] in a lengthwise direction.

[0126] However, about bit group x105-x108 and bit group **109-**117, it considers as the same processing as the gestalten 1-6 of operation. For example, it becomes un-significant when bit **117 have a value like drawing 12 .

[0127] According to the gestalt 7 of this operation, in the same bit plane, two or more groups' parallel processing becomes possible. Moreover, although the example which processes the group who followed the lengthwise direction was shown, spacing of the group of a lengthwise direction is arbitrary not only 1 but, and is good, and the same circuit as the gestalt 6 of the operation processed to juxtaposition is also good by the number of two or more arbitration. In addition, although a circuit scale will become twice if two parallel circuits are prepared, the processing engine performance also becomes twice.

[0128] The gestalt 8 of operation describes the approach in which the parallel

processing in a different bit plane is possible to the gestalt 8. pan of operation (not shown). This is using the approach stated with the gestalt 6 of operation for coincidence to a different bit plane, and the parallel processing in two or more bit planes (the number of two or more arbitration) of it becomes possible.

[0129] Moreover, although a circuit scale will become twice if the gestalt 8 of this operation also prepares two parallel circuits, the processing engine performance also becomes twice.

[0130] Furthermore, it is possible to also make coincidence perform the same circuit as the gestalt 7 of operation to a different bit plane. In addition, the parallel processing in the bit plane of the number of arbitration is possible also in this case.

[0131]

[Effect of the Invention] The processing circuit using the art of bit modeling and it concerning this invention In the art of the SHIGUNIFIKANSU propagation decoding pass (Significance propagation decoding pass) of the bit modeling of JPEG2000 The context and decision of the data which change according to the condition of the significant flag of the bit which is a processing object, and a surrounding bit group, The context and decision of a sign bit which change

according to the condition of the bit which is a processing object, and the sign bit of a surrounding bit group are generated to coincidence. Only when the value of the bit which is a processing object is 1, the context and decision of a sign bit are adopted. Update a significant flag, and when the value of the bit which is a processing object is 0, it discards. Since the value of the bit which is a processing object applies at coincidence the processing circuit of the bit modeling which updates a processed flag irrespective of 1 or 0 to four bits of the same group and processes it to juxtaposition, it can make processing speed quick.

[0132] Moreover, it sets to the art of magnitude refinement MENTOPASU (Magnitude refinement pass) of the bit modeling of JPEG2000. The 2nd bit, a processed flag, and a significant flag are referred to. it is the information on whether the bit which is a processing object is processed for the first time by magnitude refinement MENTOPASU -- significant -- When judging whether it processes or not and processing by magnitude refinement MENTOPASU Since the context and decision of a bit which are a processing object are generated, the processing circuit of the bit modeling which also updates a processed flag is applied to four bits of the same group at coincidence and it processes to

juxtaposition, processing speed can be further made quick.

[0133] Moreover, it sets to the art of the clean-up pass (Cleanup pass) of the bit modeling of JPEG2000. It judges whether all the bits of the same group who is a processing object can process collectively, when unsettled. The 1st processing circuit of bit modeling where all the same group's bit groups generate a context and decision special at the time of un-significant **, Nothing is performed about a bit [finishing / processing], but the 2nd processing circuit of the bit modeling which processes the bit of non-**** is prepared. One of the 1st processing circuit Since the 2nd processing circuit is applied to four bits of the same group at coincidence and it processes to juxtaposition, processing speed can be further made quick.

[0134] Moreover, in the art of the bit modeling of JPEG2000, since three kinds of coding pass is processed in order about the same bit plane, processing speed can be further made quick.

[0135] Moreover, in the art of the bit modeling of JPEG2000, since three groups whom the same bit plane adjoined are processed to juxtaposition for three kinds of every coding pass, processing speed can be further made quick and a circuit scale can be reduced.

[0136] Moreover, in the art of bit modeling according to claim 5, since it processes to two or more juxtaposition with the same bit plane, processing speed can be further made quick and a circuit scale can be reduced.

[0137] Moreover, in the art of one of bit modeling according to claim 5 or 6, since it processes to juxtaposition with two or more bit planes, processing speed can be further made quick.

[0138] Moreover, in the processing circuit using the art of bit modeling according to claim 1, since it has the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag and sign bit of the bit which is a processing object, and a surrounding bit group, and the register which stores the processed flag of the bit which is a processing object, processing speed can be made quick.

[0139] moreover, in the processing circuit using the art of bit modeling according to claim 2, it is the information on whether the register which stores the value of the data of the bit which is a processing object, the register which stores the significant flag of the bit which is a processing object, and a surrounding bit group, and the bit which is a processing object are processed for the first time by magnitude refinement MENTOPASU -- significant -- since it has the register

which stores the 2nd bit, processing speed can be further made quick.

[0140] moreover, the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- the data bit for a code block size, a sign bit, a processed flag, and a significant flag -- and significant -- since it has the register which stores the 2nd bit, processing speed can be made quick and a circuit scale can be reduced.

[0141] furthermore, the processing circuit using the art of either bit modeling of claims 5 and 6 and eight publications -- setting -- a data bit, a sign bit, a processed flag, and a significant flag -- and significant -- since it has the register which stores only a part for the bit which sets the 2nd bit as the object of processing, processing speed can be further made quick and a circuit scale can be reduced.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] In the bit modeling of JPEG2000 by the gestalt 1 of implementation of this invention, it is drawing showing the range of the data used when processing a certain group.

[Drawing 2] In the bit modeling of JPEG2000 by the gestalt 1 of implementation of this invention, it is drawing showing the range of the data used when processing bit O0.

[Drawing 3] In the bit modeling of JPEG2000 by the gestalt 1 of implementation of this invention, it is drawing showing the range of the data used when

processing bit O1.

[Drawing 4] In the bit modeling of JPEG2000 by the gestalt 1 of implementation of this invention, it is drawing showing the range of the data used when processing bit O2.

[Drawing 5] In the bit modeling of JPEG2000 by the gestalt 1 of implementation of this invention, it is drawing showing the range of the data used when processing bit O3.

[Drawing 6] It is drawing showing the range of the data used when processing three groups by the gestalt 5 of implementation of this invention who continued.

[Drawing 7] It is drawing showing the range of the data used when processing the group of bit group O8-O11 by the gestalt 5 of implementation of this invention.

[Drawing 8] It is drawing showing the range of the data used when processing the group of bit group O4-O7 by the gestalt 5 of implementation of this invention.

[Drawing 9] It is drawing showing the range of the data used when processing the group of bit group O0-O3 by the gestalt 5 of implementation of this invention.

[Drawing 10] In the bit modeling of JPEG2000 by the gestalt 6 of implementation of this invention, it is drawing in which having divided into a sign bit and data the data set as the object of processing, and having shown the value for every bit

plane.

[Drawing 11] In the bit modeling of JPEG2000 by the gestalt 6 of implementation of this invention, it is drawing in which having divided into a sign bit and data the data set as the object of another processing, and having shown the value for every bit plane.

[Drawing 12] In the bit modeling of JPEG2000 by the gestalt 6 of implementation of this invention, it is drawing in which having divided into a sign bit and data the data set as the object of still more nearly another processing, and having shown the value for every bit plane.

[Drawing 13] It is drawing showing the range of the data used when processing three groups by the gestalt 7 of implementation of this invention who continued to two or more juxtaposition.

[Drawing 14] In the bit modeling of conventional JPEG2000, it is drawing showing the sequence of the group who processes in a bit plane.

[Drawing 15] In the bit modeling of conventional JPEG2000, it is drawing showing the sequence of the group who processes in another bit plane.

[Description of Notations]

O 0-O3, O4-O7, O8-O11 Bit group

x0-**7 Surrounding bit group

O 0, O1, O2, O3 Each bit

x5-x8, O0-O11, **9-**12 Bit group

O 000-O011, O100-O111 Bit group

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開2003-8906

(P2003-8906A)

(43)公開日 平成15年1月10日(2003.1.10)

(51)Int.Cl. ⁷	識別記号	F I	テ-マ-ト*(参考)
H 0 4 N 1/41		H 0 4 N 1/41	B 5 C 0 5 9
H 0 3 M 7/30		H 0 3 M 7/30	Z 5 C 0 7 8
H 0 4 N 7/24		H 0 4 N 7/13	Z 5 J 0 6 4

審査請求 未請求 請求項の数11 O L (全 12 頁)

(21)出願番号 特願2001-192670(P2001-192670)

(22)出願日 平成13年6月26日(2001.6.26)

(71)出願人 00006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72)発明者 谷口 征浩

東京都千代田区丸の内二丁目2番3号 三

菱電機株式会社内

(74)代理人 100102439

弁理士 宮田 金雄 (外1名)

Fターム(参考) 5C059 KK13 MA00 MA35 PP01 UA02

5C078 BA57 CA31 DA01 DB19

5J064 AA02 BA08 BA15 BB05 BC02

BC04 BD01

(54)【発明の名称】 ビットモデリングの処理方法及びそれを用いる処理回路

(57)【要約】

【課題】 処理速度の速いビットモデリングの処理方法及びそれを用いる処理回路を得ることである。

【解決手段】 sigパスにより、処理対象であるビットと周囲のビット群の有意フラグの状態及び符号ビットの状態に応じて変化するコンテキスト及びデシジョンとを同時に生成し、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデシジョンを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃止し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループのビット00~03に同時に適用し、並列に処理する。

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								

【特許請求の範囲】

【請求項1】 J P E G 2 0 0 0 のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス (Significance propagation decoding pass) の処理方法において、処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデシジョンと、

前記処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデシジョンとを同時に生成しておき、

前記処理対象であるビットの値が1のときのみ前記符号ビットのコンテキスト及びデシジョンを採用し、前記有意フラグを更新し、

前記処理対象であるビットの値が0のときは廃棄し、前記処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項2】 J P E G 2 0 0 0 のビットモデリングのマグニチュードリファinementパス (Magnitude refinement pass) の処理方法において、

処理対象であるビットが前記マグニチュードリファinementパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、

前記マグニチュードリファinementパスで処理する場合は、前記処理対象であるビットのコンテキスト及びデシジョンを生成し、前記処理済フラグも更新するビットモデリングの処理回路を、

同一グループの4個のビットに同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項3】 J P E G 2 0 0 0 のビットモデリングのクリーンアップパス (Cleanup pass) の処理方法において、

処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、

前記同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成する前記ビットモデリングの第1の処理回路と、

処理済のビットについては何も行わず、非有意のビットについて処理を行う前記ビットモデリングの第2の処理回路とを設け、

前記第1の処理回路の1つと、前記同一グループの4個のビットに前記第2の処理回路とを同時に適用し、並列に処理することを特徴とするビットモデリングの処理方法。

【請求項4】 J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンについて、3種

類のコーディングパスを順番に処理することを特徴とするビットモデリングの処理方法。

【請求項5】 J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンの隣り合った3個のグループを、3種類のコーディングパスごとに並列に処理することを特徴とするビットモデリングの処理方法。

【請求項6】 請求項5記載のビットモデリングの処理方法において、

同一ビットプレーンで複数並列に処理することを特徴とするビットモデリングの処理方法。

【請求項7】 請求項5または6記載のいずれかのビットモデリングの処理方法において、複数のビットプレーンで並列に処理することを特徴とするビットモデリングの処理方法。

【請求項8】 請求項1記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、

処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、前記処理対象であるビットの処理済フラグを格納するレジスタとを備えることを特徴とする処理回路。

【請求項9】 請求項2記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、

処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと、

処理対象であるビットがマグニチュードリファinementパスで初めて処理されるかどうかの情報である有意第2ビットを格納するレジスタとを備えることを特徴とする処理回路。

【請求項10】 請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを格納するレジスタを備えることを特徴とする処理回路。

【請求項11】 請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、データビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを処理の対象とするビット分だけを格納するレジスタを備えることを特徴とする処理回路。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、ビットモデリングの処理方法及びそれを用いる処理回路に係り、特に J P E G 2 0 0 0 の符号化におけるビットモデリングの高速化に関するビットモデリングの処理方法及びそれを用

いる処理回路である。

【0002】

【従来の技術】J P E G 2 0 0 0 (I S O / I E C F D I S 1 5 4 4 4 - 1 で定義。以下、J P E G 2 0 0 0 と示す。)の符号化処理は、量子化係数を符号ビット及び絶対値で表す。コードブロックという単位で絶対値を表すビットを上位ビットプレーンから下位ビットプレーンの順に各ビットの上下左右の関係を調べながらコンテキスト及びデジションを生成する(以下、ビットモデリングと示す。)

【0003】また、コードブロック内の絶対値を表すビットプレーンの最上位から全てのビットが0であるビットプレーンがN0個続くときは、別に処理する。

【0004】例えば、絶対値としてMビットプレーンの精度があり、あるコードブロックにおいて、全ての絶対値の上位N0ビットが0であるようなときは、 $N = M - N0$ ビットプレーンのみビットモデリングの対象となる。

【0005】ビットモデリングの対象となる最上位のビットプレーンをMSB (Most Significant Bit プレーン、以下、MSBと示す。)、最下位のビットプレーンをLSB (Less Significant Bit プレーン、以下、LSBと示す。)と呼ぶ。

【0006】コードブロック内のデータは、MSBからLSBまでビットプレーンごとに処理して行く。各ビットプレーンにおいては縦方向に4個のビットをまとめてグループ化しておき、ラスタ順にビットモデリングを行う。

【0007】図14は横8(0~7)×縦12(0~11)のコードブロックサイズでビットモデリングを行うときのグループを処理する順番(0~23)を示す。

【0008】また、図15はビットプレーンのグループを跨ぐ個々のビット(縦方向に4個ずつのビット、例えば、横0×縦4(0~3)、横1×縦4(0~3))の処理順(0~7)を示す。

【0009】ビットモデリングにはSignificance propagation decoding pass (以下、sigパスと示す。)、Magnitude refinement pass (以下、refパスと示す。)、Cleanup pass (以下、c lnパスと示す。)という3種のコーディングパスがあり、sigパス、refパス、c lnパスの順番で処理される。

【0010】即ち、この処理は、あるビットプレーンにおいて図14、15に示した順にビットを評価して、それがsigパスで処理すべきデータであればsigパスで処理し、そうでなければ次のビットに進む。

【0011】コードブロック処理が終了すると、次にrefパスとして順番に評価して行く。最後にc lnパス

を処理する。このとき、あるビットが処理順序の早い(優先度が高い)コーディングパスで処理されると、そのビットは他のコーディングパスでは処理しない。

【0012】従って、各ビットは上記3種類の内、1種類だけのコーディングパスで処理されることになる。この処理をMSBからLSBまで繰り返すが、MSBに限りc lnパスのみで処理する。

【0013】例えば、 $N = 8$ ビットのとき、(7ビットプレーン×3コーディングパス+1コーディングパス)×(8×12)コードブロックサイズのビットモデリングの演算が必要になる。

【0014】

【発明が解決しようとする課題】上記のような従来のJ P E G 2 0 0 0のビットモデリングの処理方法では、例えば、あるコードブロックをビットモデリングするとき、1個のデータは処理するビットプレーンに応じて $(N-1) \times 3 + 1$ 倍の処理が必要になる。

【0015】これは1個のデータが、ビット精度に係わらず1回の処理だけで済む量子化等と比べて、処理速度が遅くなるという問題がある。

【0016】

【課題を解決するための手段】この発明に係るビットモデリングの処理方法及びそれを用いる処理回路は、J P E G 2 0 0 0のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス(Significance propagation decoding pass)の処理方法において、処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデジションと、処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデジションとを同時に生成しておき、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデジションを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃棄し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するものである。

【0017】また、J P E G 2 0 0 0のビットモデリングのマグニチュードリファインメントパス(Magnitude refinement pass)の処理方法において、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、マグニチュードリファインメントパスで処理する場合は、処理対象であるビットのコンテキスト及びデジションを生成し、処理済フラグも更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するものである。

【0018】また、J P E G 2 0 0 0のビットモデリングのクリーンアップパス (C l e a n u p p a s s) の処理方法において、処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成するビットモデリングの第1の処理回路と、処理済のビットについては何も行わず、非有意のビットについて処理を行うビットモデリングの第2の処理回路とを設け、第1の処理回路の1つと、同一グループの4個のビットに第2の処理回路とを同時に適用し、並列に処理するものである。

【0019】また、J P E G 2 0 0 0のビットモデリングの処理方法において、同一ビットプレーンについて、3種類のコーディングパスを順番に処理するものである。

【0020】また、J P E G 2 0 0 0のビットモデリングの処理方法において、同一ビットプレーンの隣り合った3個のグループを、3種類のコーディングパスごとに並列に処理するものである。

【0021】また、請求項5記載のビットモデリングの処理方法において、同一ビットプレーンで複数並列に処理するものである。

【0022】また、請求項5または6記載のいずれかのビットモデリングの処理方法において、複数のビットプレーンで並列に処理するものである。

【0023】また、請求項1記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、処理対象であるビットの処理済フラグを格納するレジスタとを備えるものである。

【0024】また、請求項2記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットを格納するレジスタとを備えるものである。

【0025】また、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを格納するレジスタを備えるものである。

【0026】さらに、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、データビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを処理の対象とするビット分だけを格納するレジスタを備えるものである。

【0027】

【発明の実施の形態】実施の形態1. 以下、この発明について説明する。図1は実施の形態1によるJ P E G 2 0 0 0のあるビットプレーンにおけるグループのビット群00~03を処理する場合を示した図である。図1を参照して、ビット群00~03で示すグループをs i gパスで処理する場合、その周囲のビットの内、×で示すビットについての情報が必要になる。

【0028】図2は図1における00のビットを処理する場合に必要なビットを示した図である。図2を参照して、×0、×1、×2、×3、×5については同一コーディングパスの処理においてビット00より先に処理されるビットを示している。△4、△6、△7についてはビット00より後に処理されるビットを示している。

【0029】s i gパスにおいてビットが処理対象となる条件は、そのビットがi n s i g n i f i c a n c e (以下、非有意と示す。)であり、周囲のビットが1個以上s i g n i f i c a n c e (以下、有意と示す。)であることである。

【0030】従って、処理対象であるビット及び周囲のビットが有意であるかどうか (以下、有意フラグと示す。)の情報が必要になる。有意フラグはレジスタに格納しておく。

【0031】ビット00が非有意で、周囲のビット群×0~△7の内、1個以上が有意であればs i gパスで処理することとなる。

【0032】周囲のビット群×0~△7の有意フラグの状態に応じてコンテキスト (以下、データのコンテキストと示す。)を生成し、ビット00の値そのものをデシジョン (以下、データのデシジョンと示す。)とする。

【0033】ビット00の値が1であれば、そのデータが非有意から有意になるので、有意フラグを更新して、さらに符号ビットについてもビットモデリングを行う。

【0034】符号ビット処理には、ビット00の符号ビット及び周囲のビット群×0~△7の符号ビットが必要になる。

【0035】周囲のビット群×0~△7の符号ビットの状態に応じてコンテキスト (以下、符号ビットのコンテキストと示す。)を生成し、ビット00の符号ビット及びコンテキストに応じた値のX O R演算の結果をデシジョン (以下、符号ビットのデシジョンと示す。)とする。

【0036】データのコンテキスト/デシジョン及び符号ビットのコンテキスト/デシジョンは同時に生成しておき、必要なとき (ビット00の値が1)のみ符号ビットのコンテキスト/デシジョンを採用する。

【0037】また、必要のないとき (ビット00の値が0)は廃棄すれば並列に処理することができる。

【0038】また、ビット00の値が1であるか0であるかに係わらず、s i gパスで処理されたときは、その

ビットプレーンにおいて処理済であるという情報（以下、処理済フラグと示す。）を更新する。

【0039】あるビットをsigパスで処理するときは、そのビットの値、そのビットと周囲のビットの有意フラグ及び符号ビットとを参照する。

【0040】即ち、ビット0の値、ビット0と周囲のビット群×0～△7の有意フラグ及び符号ビットとを格納するレジスタが必要になる。

【0041】また、ビット0の処理済フラグのためのレジスタについても用意する必要がある。

【0042】図3は図1における○1のビットを処理する場合に必要となるビットを示した図である。図3を参照して、ビット01をsigパスで処理するかどうかの判断には、周囲のビット群×0～△7の有意フラグを参照する。

【0043】また、ビット×1の有意フラグは図2のビット00の有意フラグと同等なので、ビット00の処理時に更新される可能性があり、ビット00の処理が終了するまでビット01の処理は開始できないはずであるが、ビット01を処理するときにビット×1の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0044】即ち、ビット×1が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット×1及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット00及びビット01とを同時に処理する。

【0045】図4は図1における○2のビットを処理する場合に必要となるビットを示した図である。図4を参照して、ビット02をsigパスで処理するかどうかの判断には、周囲のビット群×0～△7の有意フラグを参照する。

【0046】ビット×1の有意フラグは図3のビット01の有意フラグと同等なので、ビット01の処理時に更新される可能性があり、ビット01の処理が終了するまでビット02の処理は開始できないはずであるが、ビット02を処理するときにビット×1の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0047】即ち、ビット×1が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット×1及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット01及びビット02とを同時に処理する。

【0048】図5は図1における○3のビットを処理する場合に必要となるビットを示した図である。図5を参

照して、ビット03をsigパスで処理するかどうかの判断には、周囲のビット群×0～△7の有意フラグを参照する。

【0049】ビット×1の有意フラグは図4のビット02の有意フラグと同等なので、ビット02の処理時に更新される可能性があり、ビット02の処理が終了するまでビット03の処理は開始できないはずであるが、ビット03を処理するときにビット×1の有意フラグが更新されるかどうか解っていれば、同時に処理することが可能となる。

【0050】即ち、ビット×1が非有意のとき、その値が1でありその周囲のビット群が1個以上有意である場合に、直前のsigパスの処理で有意になると判断し、ビット×1及びその周囲のビット群の有意フラグ及びそのビットの値とを同時に参照して更新後の有意フラグとすることで、ビット02及びビット03とを同時に処理する。

【0051】この実施の形態1によると、ビット群00～03のグループの並列処理が可能となり、1ビットのデータについてはsigパスで処理するかどうか判断し、コンテキスト/デシジョンのセットを0/1/2個出力する回路を構成することが可能となる。

【0052】また、1ビットのデータを処理する回路を4個並列に並べてそれぞれビット00、01、02、03について処理することで、1個のグループ（4個のデータ）について、0～8個のコンテキスト/デシジョンのセットを出力する回路を構成することが可能となる。

【0053】実施の形態2. 次に、実施の形態2による並列処理方法について述べる。これは、図1～5をrefパスで処理する実施の形態であるため、実施の形態1における図1～5と同様の図面を用いて説明する。

【0054】まず、図1のビット群00～03で示すグループをrefパスで処理するとき、その周囲のグループの内、×で示すビットについての情報が必要になる。このグループをさらに各ビット毎に分けると図2～5に示す各ビット00、01、02、03について、周囲8ビットの情報が必要となる。

【0055】refパスでのビットの処理条件は、そのビットがsigパスで処理されてなく、有意であることが必要である。従って、処理対象であるビットが処理済であるか、また、有意であるかどうかの情報が必要となる。

【0056】図2のビット00において、未処理で有意であればrefパスにより処理することになる。初めてビット00がrefパスで処理されるとき、周囲のビット群×0～△7の有意フラグの状態に応じてデータのコンテキストを生成し、そうでないときは有意フラグの状態によらず固定のデータのコンテキストを生成する。

【0057】いずれのときもビット00の値そのものをデータのデシジョンとし、refパスで処理されるとき

10

20

30

40

50

は処理済フラグを更新し、有意フラグ、処理済フラグはsigパスと同様のものを使用する。さらに、初めてビット00がrefパスで処理されるかどうかの情報(以下、有意第2ビットと示す。)はレジスタに格納しておく。

【0058】図3においてビット01をrefパスで処理するかどうかの判断は、ビット01の処理済フラグ及び有意フラグとを参照する。同様に、図4、5についてもそれぞれビット02、ビット03の処理済フラグ及び有意フラグとを参照する。

【0059】また、図2～5に示す各ビット00、01、02、03を処理するには図1に示すビット群00～03の値、処理済フラグ及び有意フラグと、ビット0及びビットxとの有意フラグが必要になる。

【0060】この実施の形態2によると、refパスは有意フラグを更新することがないので、ビット群00～03のグループの並列処理が可能となり、1ビットのデータについてはrefパスで処理するかどうか判断し、コンテキスト/デシジョンのセットを0/1個出力する回路を構成することが可能となる。

【0061】また、1ビットのデータを処理する回路を4個並列に並べてそれぞれビット00、01、02、03について処理することで、1個のグループ(4個のデータ)について、0～4個のコンテキスト/デシジョンのセットを出力する回路を構成することが可能となる。

【0062】実施の形態3。実施の形態3による並列処理方法について述べる。これは、図1～5をc1nパスで処理する実施の形態であるため、実施の形態1における図1～5と同様の図面を用いて説明する。

【0063】まず、図1のビット群00～03で示すグループをc1nパスで処理するとき、その周囲のグループの内、xで示すビットについての情報が必要になる。このグループをさらに各ビット毎に分けると図2～5に示す各ビット00、01、02、03について、周囲8ビットの情報が必要となる。

【0064】c1nパスでのビットの処理条件は、そのビットが未処理であることである。但し、MSBではc1nパスしか処理しないので全てのビットが処理対象となる。従って、処理対象であるビットが処理済であるかどうかの情報が必要となる。

【0065】また、c1nパスでは図2～5に示す各ビット00、01、02、03において、全て未処理のとき、まとめて処理できるかどうかを判断する。図2～5に示す各ビット00、01、02、03のそれぞれにおいて、周囲のビット群x0～x7が全て非有意のときは特別なコンテキスト(以下、ランレングスのコンテキストと示す。)を生成する。

【0066】また、同時に図2～5に示す各ビット00、01、02、03の値が全て0であればデシジョン(以下、ランレングスのデシジョンと示す。)0を生成

し、そのグループの処理を終了する。

【0067】また、図2～5に示す各ビット00、01、02、03の値が1つでも1であれば、ランレングスのデシジョンは1とする。このとき図2～5に示す各ビット00、01、02、03の内、最初に1の値を持つビットの位置を2ビットのデータで表す。

【0068】即ち、ビット00が最初の1であれば00、ビット01であれば01、以下、同様10、11とする。これらの2つのデータはランレングスに続くコンテキスト(以下、UNIFORMのコンテキストと示す。)とともに生成され、それぞれデシジョン(以下、UNIFORMのデシジョンと示す。)として出力される。

【0069】ランレングスのコンテキストだけが出力されるときは、全て非有意のままであるので有意フラグの更新は行われない。

【0070】UNIFORMのコンテキストも出力されるとき、UNIFORMのデシジョンが示す位置のビットは非有意から有意になるので、そのビットの有意フラグを更新する。

【0071】但し、UNIFORMのコンテキスト自体が、そのデシジョンの示す位置のビットが1であることを表すので、sigパスと同様にそのビットの符号ビットのコンテキストを生成する。

【0072】c1nパスでの符号ビットのコンテキストの生成方法はsigパスと同様であり、UNIFORMのデシジョンが11のときは符号ビットのみの処理で終了となるが、それ以外のときは残るビットを1ビットづつ処理する。

【0073】残るビットの処理の方法はそのビットの周囲のビットが全て非有意でも処理することを除いて、sigパスと同様である。また、UNIFORMのデシジョンが示す位置より前のビットは非有意のままであるので、有意フラグの更新は行わない。

【0074】この実施の形態3によると、ランレングスのコンテキストの処理対象とならないときは、そのグループ中のビットの内、未処理のものを順番に処理する。また、UNIFORMのコンテキスト以後のビット、またはランレングスのコンテキストの処理対象とならないときの未処理のビットについては、sigパスと同様の処理を行うためsigパスと同様の回路を構成することが可能となる。

【0075】また、sigパスでは各ビットが独立したものを並列に処理することで0～8個のコンテキスト/デシジョンのセットを出力する回路を構成しているが、c1nパスによると、さらに0ランレングスのコンテキスト/デシジョン、UNIFORMのコンテキスト/デシジョンを処理する回路を並列に処理させ、
・0ランレングスのコンテキストを含む場合
→UNIFORMのコンテキストを含むか

→UNIFORMのコンテキスト以後のビット数
 ・0ランレングスのコンテキストを含まない場合
 →c l nパスで処理するビット数
 を考慮して、0ランレングス、UNIFORM、各ビ
 ットのコンテキスト/デシジョンを計算することで0~1
 0個のコンテキスト/デシジョンのセットを出力する回
 路を構成することが可能となる。

【0076】実施の形態4. 実施の形態1~3について
 は、各コーディングパスについて個別の回路構成につ
 いて述べたものであるが、実施の形態4では同一ビット
 プレーン内の同一グループについて実施の形態1~3の回
 路で同時に処理するものである。

【0077】この場合、3種類のコーディングパスで情
 報を共有する必要があるため、コードブロックサイズ分
 の有意フラグ、有意第2ビット、処理済フラグをレジス
 タに格納しておく必要がある。

【0078】また、符号ビット、処理対象となるビット
 プレーンのデータビットについては、コードブロックサ
 イズ分、または実際に処理対象となる部分を各コーディ
 ングパス毎に読み出す場合はその分だけレジスタに格納
 しておく必要がある。

【0079】この実施の形態4によると、実施の形態1
 ~3に比べ処理速度を速くすることができる。

【0080】実施の形態5. 図6は実施の形態5による
 3個の連続したグループを処理する場合に使用するデー
 タの範囲を示す図である。実施の形態4では、同一ビッ
 トプレーン内の同一グループを3種類のコーディングパ
 スで同時に処理する方法であった。

【0081】しかし、コードブロック内のあるビットプ
 レーンについては、s i gパス→r e fパス→c l nパ
 スの順で処理して行くが、各ビットは1個のコーディ
 ングパスだけで処理されることや、s i gパスあるいはc
 l nパス等複数のパスで有意フラグを更新すること等の
 制約により、正確なビットモデリングを得ようとすれ
 ば、各コーディングパスを順番に処理する必要がある。

【0082】つまり、あるビットプレーンにおいて、4
 個のビットを同時に処理しているとはいえ実際には3種
 類のコーディングパスで順番に処理することになり、回
 路としての動作周波数を低く押さえる必要がある。これ
 を解決するために実施の形態5では、横方向に連続した
 グループ(ビット群00~03、ビット群04~07、
 ビット群08~011)をそれぞれ別のコーディングパ
 スで処理する方法をとっている。

【0083】図7はビット群08~011のグループを
 処理する場合に使用するデータの範囲を示す図である。
 また、図8はビット群04~07のグループを処理する
 場合に使用するデータの範囲を示す図である。また、図
 9はビット群00~03のグループを処理する場合に使用
 するデータの範囲を示す図である。

【0084】s i gパスは非有意であり、かつ周囲のビ

ットに1個以上有意であるビットが存在するときに処理
 するコーディングパスであり、最初に実行され、他の2
 個のパスからは何の制約も受けずに処理することができ
 る。

【0085】図6で示す3個のグループではビット群0
 8~011を処理するコーディングパスであり、図7が
 そのデータの使用範囲である。ビット群08~011を
 s i gパスで処理するので、条件によりビット群08~
 011の処理済フラグ、有意フラグが更新される。

【0086】図6~9におけるビット群×5~×8、ビ
 ット群00~03、ビット群04~07はs i gパスに
 より既に処理済である。

【0087】次に、r e fパスは既に有意であるデー
 タを処理するコーディングパスであり、処理時に有意で
 あるかどうかを考慮する必要がある。

【0088】そのビットが既に有意であるかどうかは有
 意フラグを参照するが、そのビットが同じビットプレー
 ンのs i gパスで処理され有意になったときは処理でき
 ない。

【0089】即ち、s i gパスで処理されたかどうかを
 判断する必要があるため、s i gパスと同時に処理する
 ことはできない。

【0090】図6で示す3個のグループではビット群0
 4~07を処理するコーディングパスであり、図8がそ
 のデータの使用範囲である。ビット群04~07をr e f
 パスで処理するので、ビット群04~07の処理済フラ
 グ、有意フラグを参照し、条件により処理済フラグを
 更新する。

【0091】s i gパスでビット群08~011、r e f
 パスでビット群04~07を処理することで、r e f
 パスではs i gパスで更新した後のビット群04~07
 のフラグを参照することができる。

【0092】また、この実施の形態5では、回路規模を
 最小とするためs i gパスとr e fパスで処理するグル
 ープは横方向に1グループずらした構成を示したが、2
 グループ以上ずらしてもよい。

【0093】図6、8、9におけるビット群5×~×
 8、ビット群00~03はr e fパスにより既に処理済
 である。

【0094】また、c l nパスは非有意であり、s i g
 パスで処理されなかったデータを処理するコーディン
 グパスである。この実施の形態5では、有意なデータはr
 e fパスで既に処理されており、非有意なデータの内s
 i gパスで処理されなかったデータというのは、未処理
 のデータ全てであるので、処理済フラグを参照し、未処
 理のデータのみを処理する。

【0095】処理対象となるビット及び周囲ビットの有
 意フラグを参照する必要があるため、s i gパスと同時
 に処理することはできない。

【0096】図6で示す3個のグループではビット群0

0~03を処理するコーディングパスであり、図9がそのデータの使用範囲である。ビット群00~03をc l nパスで処理するので、ビット群04~07の有意フラグ、ビット群00~03の処理済フラグを参照し、条件により有意フラグを更新する。

【0097】また、図6、9におけるビット群×5~×8はc l nパスにより既に処理済である。

【0098】s i gパスでビット群08~011、r e fパスでビット群04~07、c l nパスでビット群00~03を処理することで、c l nパスではs i gパス及びr e fパスで更新した後のビット群00~03、ビット群04~07のフラグを参照することができる。

【0099】また、この実施の形態5では、回路規模を最小とするためs i gパスとr e fパスで処理するグループは横方向に1グループずらした構成を示したが、s i gパス、r e fパス及びc l nパスで処理するグループ数は任意でよい。

【0100】上記構成で3種類のコーディングパスを並列に処理するとき、実施の形態1~4と同様の方法とあわせて、4個のデータを3種類のコーディングパスだけ並列に処理できる。

【0101】また、図6においてs i gパスでビット群△9~△12、r e fパスでビット群08~011、c l nパスでビット群04~07というように処理するグループをずらすことで連続に処理することができる。

【0102】各コーディングパスは0~10個のコンテキスト/デシジョンのセットを出力するが、これらのデータはコーディングパス毎に格納する必要がある。

【0103】この実施の形態5によると、コードブロックサイズ分の有意フラグ、有意第2ビットの情報をレジスタに格納しておく必要があるが、3種類のコーディングパスを1度に処理するため、符号ビット、処理対象となるビットプレーンのデータビット、処理済フラグについては実際に処理する場合に処理の対象となっている部分だけレジスタに格納しておけばよい。

【0104】また、図6のように、各コーディングパスで処理するグループの間隔を1グループにした場合、処理の対象となっている部分というのは、符号ビットでは○△×の30ビット分、データビット及び処理済フラグでは×5~×8、00~011、△9~△12の20ビット分になる。

【0105】従って、実施の形態1~4に比べ、高速に処理でき、回路規模を縮小できる。

【0106】実施の形態6。次に、実施の形態6による並列処理方法について述べる。これは、図6における有意フラグ及び有意第2ビットを処理の対象とする部分(○△×の30ビット分(有意フラグ)、×5~×8、00~011、△9~△12の20ビット分(有意第2ビット))だけのレジスタ量で実行する方法である。

【0107】有意フラグがコードブロックサイズ分必要

であるのは、MSBからLSBまでビットモデリングをしている間、情報を共有する必要があるため、保持する必要があるからである。

【0108】しかし、あるビットプレーンの処理においては、その直前のビットプレーンでの処理後の状態が解ればよく、直前のビットプレーンまで処理した後の状態を予め計算することができれば、ビットプレーン間で情報を共有する必要がなくなる。

【0109】実施の形態6では、処理しようとするデータの値を調べ、処理対象となるビットプレーンより上位に値が1であるビットがあるかどうかで判断する方法をとる。

【0110】図10は、ビットモデリングで処理しようとするある1つのデータを符号ビット、及び絶対値に分け、その値を示した図である。図10を参照して、MSBから処理対象であるビットプレーンより1個上のビットまでの値は全て0であるので、このデータのs i gパス処理前の有意フラグは非有意とする。

【0111】また、図11では、MSBから処理対象であるビットプレーンより1個上のビットまでの値が1であるので、このデータのs i gパス処理前の有意フラグは有意とする。

【0112】これをビットプレーン毎にデータビットを読み出すときに実行すれば、ビットプレーン間で情報を共有する必要がなくなるため、有意フラグのレジスタ数を削減することができる。

【0113】従って、有意フラグは図6に示す○△×の30ビット分だけレジスタを持てばよいことになる。

【0114】また、有意第2ビットも同様に、初めてr e fパスで処理するビットプレーン、即ち、有意になった次のビットプレーンを予め計算しておくことにより、ビットプレーン間で情報を共有する必要がなくなる。

【0115】この実施の形態6では、処理しようとするデータを調べ、処理対象となるビットプレーンより1ビット上位の値が1で、それより上位のビットが全て0であるかどうかで判断する。

【0116】図10を例にとると、処理対象のビットプレーンより上位のビットの値が全て0であるので有意第2ビットにはならない。また、図11では、処理対象するビットプレーンの1ビット上位の値が1で、それより上位ビットの値が全て0であるので有意第2ビットになる。

【0117】これをビットプレーン毎にデータビットを読み出すときに実行すれば、ビットプレーン間で情報を共有する必要がなくなるため、有意第2ビットのレジスタ数を削減することができる。

【0118】従って、有意第2ビットは図6に示す×5~×8、00~011、△9~△12の20ビット分だけ持てばよいことになる。

【0119】実施の形態6は、同一のビットプレーンに

において3種類のコーディングパスを並列に処理する回路であり、以下の実施の形態の基本回路とする。

【0120】実施の形態7. この実施の形態7は、同一のビットプレーンにおいて実施の形態6の回路を複数使用する処理方法である。図13は3個の連続したグループを複数並列に処理する場合に使用するデータの範囲を示す図である。これはビット群0000~0011、0100~0111について2つの実施の形態6で述べた回路を使用するものである。

【0121】図13を参照して、ビット群0000~0011については実施の形態6と同様の回路を使用してそのまま処理する。また、ビット群100~104は本来ビット群0100~0111よりも前に処理されるべきデータであるので有意フラグを予め計算しておく必要がある。

【0122】実施の形態6では、処理対象となるビットプレーンの1ビット上位のビットプレーンまでの有意フラグの計算方法を示したが、この実施の形態7では、さらに縦方向に1段上のグループの処理後の情報が必要になる。これは実施の形態6で示した方法に加えて、処理対象のビットプレーンのデータも考慮することで計算できる。

【0123】以下に、図13の100が、図10~12のような値を持っている場合について述べる。

【0124】図10では、MSBから処理対象となるビットプレーンまで値が全て0なので非有意である。図11では、MSBから処理対象となるビットプレーンの1ビット上位までに値が1であるビットがあるので有意である。

【0125】また、図12では、MSBから処理対象となるビットプレーンまでに値が1であるビット（この場合は、処理対象のビットプレーン）があるので有意である。このように、縦方向に1段上のグループまで、有意フラグを計算することが可能となる。

【0126】但し、ビット群105~108、ビット群109~117については実施の形態1~6と同様の処理とする。例えば、ビット117が図12のような値を持っていた場合、非有意となる。

【0127】この実施の形態7によると、同一のビットプレーンにおいて複数のグループの並列処理が可能となる。また、縦方向に連続したグループを処理する例を示したが、縦方向のグループの間隔は1に限らず任意でよく、並列に処理する実施の形態6と同様の回路も2個以上の任意の数でよい。尚、並列回路を2個設ければ、回路規模は2倍となるが、処理性能も2倍となる。

【0128】実施の形態8. さらに、実施の形態8では異なるビットプレーンでの並列処理が可能な方法について述べる（図示せず。）。これは実施の形態6で述べた方法を異なるビットプレーンに対して同時に使用することで、複数のビットプレーン（2以上の任意の数）での

並列処理が可能となる。

【0129】また、この実施の形態8も並列回路を2個設ければ、回路規模は2倍となるが、処理性能も2倍となる。

【0130】さらに、実施の形態7と同様の回路を異なるビットプレーンに対して同時に実行させることも可能である。尚、この場合も任意の数のビットプレーンでの並列処理が可能である。

【0131】

【発明の効果】この発明に係るビットモデリングの処理方法及びそれを用いる処理回路は、JPEG2000のビットモデリングのシグニフィカンスプロパゲーションデコーディングパス（Significance propagation decoding pass）の処理方法において、処理対象であるビット及び周囲のビット群の有意フラグの状態に応じて変化するデータのコンテキスト及びデシジョンと、処理対象であるビット及び周囲のビット群の符号ビットの状態に応じて変化する符号ビットのコンテキスト及びデシジョンとを同時に生成しておき、処理対象であるビットの値が1のときのみ符号ビットのコンテキスト及びデシジョンを採用し、有意フラグを更新し、処理対象であるビットの値が0のときは廃棄し、処理対象であるビットの値が1か0に係わらず、処理済フラグを更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するので、処理速度を速くすることができる。

【0132】また、JPEG2000のビットモデリングのマグニチュードリファインメントパス（Magnitude refinement pass）の処理方法において、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットと処理済フラグ及び有意フラグとを参照して、処理するかどうかを判断し、マグニチュードリファインメントパスで処理する場合は、処理対象であるビットのコンテキスト及びデシジョンを生成し、処理済フラグも更新するビットモデリングの処理回路を、同一グループの4個のビットに同時に適用し、並列に処理するので、さらに処理速度を速くすることができる。

【0133】また、JPEG2000のビットモデリングのクリーンアップパス（Cleanup pass）の処理方法において、処理対象である同一グループのビットが全て未処理のとき、まとめて処理できるかどうかを判断し、同一グループのビット群が全て非有意のときは特別なコンテキスト及びデシジョンを生成するビットモデリングの第1の処理回路と、処理済のビットについては何も行わず、非有意のビットについて処理を行うビットモデリングの第2の処理回路とを設け、第1の処理回路の1つと、同一グループの4個のビットに第2の処理回路とを同時に適用し、並列に処理するので、さらに

処理速度を速くすることができる。

【0134】また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンについて、3種類のコーディングパスを順番に処理するので、さらに処理速度を速くすることができる。

【0135】また、J P E G 2 0 0 0 のビットモデリングの処理方法において、同一ビットプレーンの隣り合った3つのグループを、3種類のコーディングパスごとに並列に処理するので、さらに処理速度を速くことができ、回路規模を縮小することができる。

【0136】また、請求項5記載のビットモデリングの処理方法において、同一ビットプレーンで複数並列に処理するので、さらに処理速度を速くことができ、回路規模を縮小することができる。

【0137】また、請求項5または6記載のいずれかのビットモデリングの処理方法において、複数のビットプレーンで並列に処理するので、さらに処理速度を速くすることができる。

【0138】また、請求項1記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグ及び符号ビットを格納するレジスタと、処理対象であるビットの処理済フラグを格納するレジスタとを備えるので、処理速度を速くすることができる。

【0139】また、請求項2記載のビットモデリングの処理方法を用いる処理回路において、処理対象であるビットのデータの値を格納するレジスタと、処理対象であるビット及び周囲のビット群の有意フラグを格納するレジスタと、処理対象であるビットがマグニチュードリファインメントパスで初めて処理されるかどうかの情報である有意第2ビットを格納するレジスタとを備えるので、さらに処理速度を速くすることができる。

【0140】また、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、コードブロックサイズ分のデータビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを格納するレジスタを備えるので、処理速度を速くことができ、回路規模を縮小することができる。

【0141】さらに、請求項5、6、8記載のいずれかのビットモデリングの処理方法を用いる処理回路において、データビット、符号ビット、処理済フラグ、有意フラグ及び有意第2ビットとを処理の対象とするビット分だけを格納するレジスタを備えるので、さらに処理速度を速くことができ、回路規模を縮小することができる。

【図面の簡単な説明】

【図1】 この発明の実施の形態1によるJ P E G 2 0 0 0 のビットモデリングにおいて、あるグループを処理する場合に使用するデータの範囲を示す図である。

【図2】 この発明の実施の形態1によるJ P E G 2 0 0 0 のビットモデリングにおいて、ビット00を処理する場合に使用するデータの範囲を示す図である。

【図3】 この発明の実施の形態1によるJ P E G 2 0 0 0 のビットモデリングにおいて、ビット01を処理する場合に使用するデータの範囲を示す図である。

【図4】 この発明の実施の形態1によるJ P E G 2 0 0 0 のビットモデリングにおいて、ビット02を処理する場合に使用するデータの範囲を示す図である。

【図5】 この発明の実施の形態1によるJ P E G 2 0 0 0 のビットモデリングにおいて、ビット03を処理する場合に使用するデータの範囲を示す図である。

【図6】 この発明の実施の形態5による3つの連続したグループを処理する場合に使用するデータの範囲を示す図である。

【図7】 この発明の実施の形態5によるビット群08～011のグループを処理する場合に使用するデータの範囲を示す図である。

【図8】 この発明の実施の形態5によるビット群04～07のグループを処理する場合に使用するデータの範囲を示す図である。

【図9】 この発明の実施の形態5によるビット群00～03のグループを処理する場合に使用するデータの範囲を示す図である。

【図10】 この発明の実施の形態6によるJ P E G 2 0 0 0 のビットモデリングにおいて、処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図11】 この発明の実施の形態6によるJ P E G 2 0 0 0 のビットモデリングにおいて、別の処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図12】 この発明の実施の形態6によるJ P E G 2 0 0 0 のビットモデリングにおいて、さらに別の処理の対象となるデータを符号ビットとデータに分けてビットプレーン毎に値を示した図である。

【図13】 この発明の実施の形態7による3つの連続したグループを複数並列に処理する場合に使用するデータの範囲を示す図である。

【図14】 従来のJ P E G 2 0 0 0 のビットモデリングにおいて、ビットプレーン内で処理するグループの順番を示す図である。

【図15】 従来のJ P E G 2 0 0 0 のビットモデリングにおいて、別のビットプレーン内で処理するグループの順番を示す図である。

【符号の説明】

00～03、04～07、08～011 ビット群

×0～△7 周囲のビット群

00、01、02、03 各ビット

×5～×8、00～011、△9～△12 ビット群

〇〇〇〇～〇〇111、〇100～〇111 ビット群

【図1】

	0	1	2	3	4	5	6	7
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								

【図2】

	×0	×1	×2
×3		〇0	△4
×5		△6	△7

【図3】

	×0	×1	△2
×3		〇1	△4
×5		△6	△7

【図4】

	×0	×1	△2
×3		〇2	△4
×5		△6	△7

【図5】

	×0	×1	△2
×3		〇3	△4
△5		△6	△7

【図8】

	×1	×2	×3
〇0		〇4	〇8
〇1		〇5	〇9
〇2		〇6	〇10
〇3		〇7	〇11
△14		△15	△16

【図9】

	×0	×1	×3
×5		〇0	〇4
×6		〇1	〇5
×7		〇2	〇6
×8		〇3	〇7
△13		△14	△15

【図6】

	×0	×1	×2	×3	×4
×5		〇0	〇4	〇8	△9
×6		〇1	〇5	〇9	△10
×7		〇2	〇6	〇10	△11
×8		〇3	〇7	〇11	△12
△13		△14	△15	△16	△17

【図7】

	×3	×4	×5
〇4		〇8	△9
〇5		〇9	△10
〇6		〇10	△11
〇7		〇11	△12
△15		△16	△17

【図12】

【図10】

1	符号ビット
0	MSB
0	
0	
0	処理するビットプレーン
0	
1	
0	
0	LSB

【図11】

0	符号ビット
0	MSB
0	
1	
0	処理するビットプレーン
0	
0	
0	
0	LSB

0	符号ビット
0	MSB
0	
0	
1	処理するビットプレーン
0	
0	
0	
0	LSB

【図13】

×000	×001	×002	×003	×004					
×005	○000	○004	○008	△009					
×006	○001	○005	○009	△010					
×007	○002	○006	○010	△011					
×008	○003	○007	○011	△012	×100	×101	×102	×103	×104
△013	△014	△015	△016	△017	×105	○100	○104	○108	△109
					×106	○101	○105	○109	△110
					×107	○102	○106	○110	△111
					×108	○103	○107	○111	△112
					△113	△114	△115	△116	△117

【図15】

	0	1	2	3	4	5	6	7
0	0	4						
1	1	5						
2	2	6						
3	3	7						
4								
5								
6								
7								
8								
9								
10								
11								

【図14】

	0	1	2	3	4	5	6	7
0								
1	0	1	2	3	4	5	6	7
2								
3								
4								
5								
6	8	9	10	11	12	13	14	15
7								
8								
9								
10	16	17	18	19	20	21	22	23
11								